Cafe Scheduling Optimization Simulation
Initial Implementation
Repository: https://github.com/Baesiann/CS4632_Kenneth_Burke

Kenneth Burke
Department of Computer Science
Kennesaw State University
Kennesaw, Georgia 30144
Email: kburke36@students.kennesaw.edu

1. Project Board

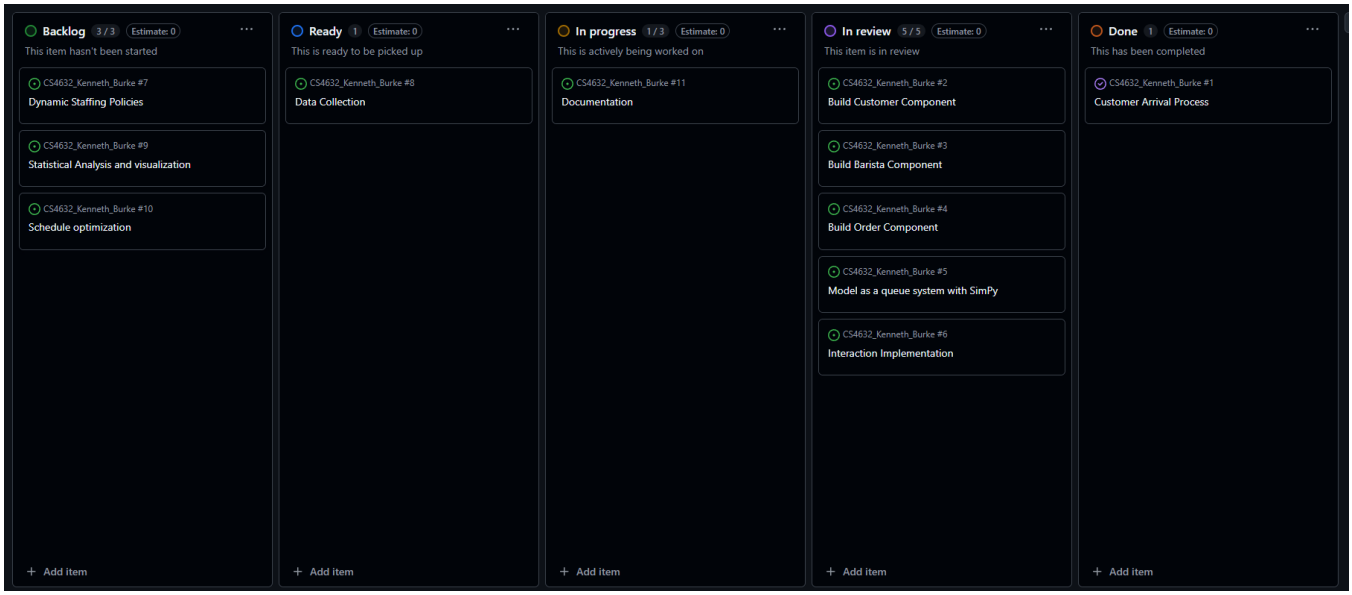

Figure 1. Project Board using GitHub Projects

1.1. Completed/In Progress

Completed.

- Project Structure Setup
- Implementation of Customer Arrival Process
- Customer life cycle
- Data collection module
- CSV exporting

In-Progress.

- Barista queue handling
- Integration of order types and pricing

1.2. To-Do

- Creating a GUI
- Iteration over multiple days
- Creating data visualization
- Schedule optimization

1.3. Implementation Evidence

```
C:\Fall2025\CS4632\cafe-simulation\src>python main.py
   CustomerID    OrderType  Price  ArrivalTime  ...  EndService  WaitTime  ServiceTime  TotalTime
0           1       coffee    3.0           12  ...   13.347754  0.000000     1.347754   1.347754
1           2        latte    4.5           14  ...   16.754364  0.000000     2.754364   2.754364
2           4       coffee    3.0           19  ...   20.768917  0.000000     1.768917   1.768917
3           5       coffee    3.0           19  ...   22.226920  1.768917     1.458002   3.226920
4           3  frappuccino    6.0           19  ...   25.619070  0.000000     6.619070   6.619070

[5 rows x 9 columns]
Total revenue: $312.00
Average wait time: 0.20 minutes
Throughput: 0.17 customers/min
[+] Data saved to C:\Fall2025\CS4632\cafe-simulation\data\simulation_results.csv
```

Figure 2. Console output of simulation

The console output is proof that I have a working simulation, that the customer life cycle is implemented with metrics in the below figure attached to each customer.

	A	B	C	D	E	F	G	H	I
1	Customer	OrderType	Price	ArrivalTim	StartServic	EndServic	WaitTime	ServiceTin	TotalTime
2	1	coffee	3	12	12	13.34775	0	1.347754	1.347754
3	2	latte	4.5	14	14	16.75436	0	2.754364	2.754364
4	4	coffee	3	19	19	20.76892	0	1.768917	1.768917
5	5	coffee	3	19	20.76892	22.22692	1.768917	1.458002	3.22692
6	3	frappucci	6	19	19	25.61907	0	6.61907	6.61907
7	6	coffee	3	79	79	81.22329	0	2.223289	2.223289
8	8	coffee	3	84	84	85.27536	0	1.275359	1.275359
9	7	frappucci	6	83	83	87.2112	0	4.2112	4.2112
10	9	latte	4.5	85	85.27536	89.02623	0.275359	3.750876	4.026235
11	11	coffee	3	103	103	105.0188	0	2.018807	2.018807
12	10	coffee	3	103	103	105.109	0	2.109045	2.109045
13	13	latte	4.5	108	108	110.6226	0	2.622608	2.622608
14	12	latte	4.5	107	107	111.056	0	4.055992	4.055992
15	14	coffee	3	111	111	113.7163	0	2.7163	2.7163

Figure 3. CSV output of simulation

The CSV output highlights the collected metrics, each customer has an associated ID, tracked metrics include order type, time of arrival, time service starts, time service ends, time spent in queue, total time of service, and the total time of the moment the customer entered to when served.

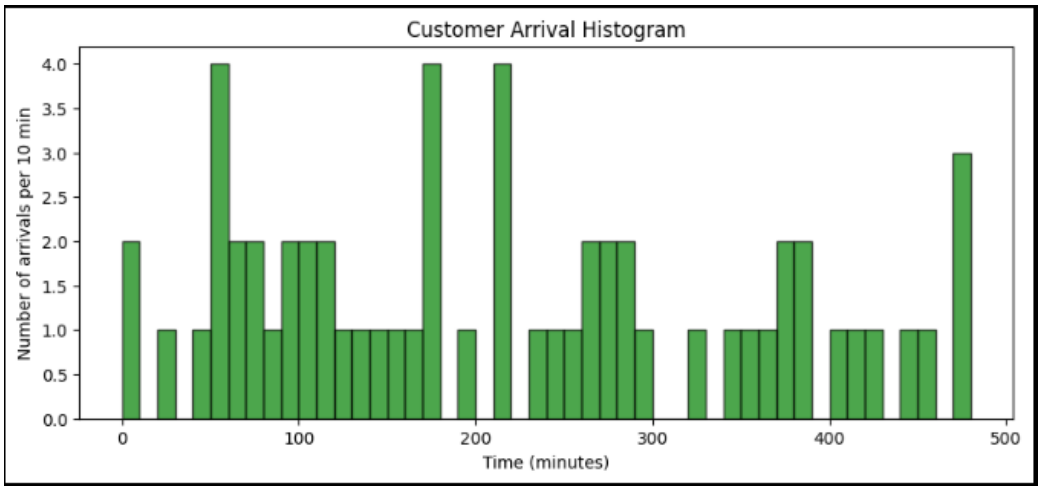

Figure 4. Arrival Histogram

The figure above is the results of a doubly stochastic model that I used to simulate customer arrivals. The chart shows reflection of real operations, as there are ”rush hours” of higher volumes of customers.

1.4. Status Summary

Since Milestone 1, the project has progressed from an initial proposed plan to a functional simulation with customer arrivals and queueing models implemented. The simulation uses a doubly stochastic Poisson process (Cox process) to model the arrival intensity throughout the 8 hours of operation. The arrival process has been integrated into SimPy, allowing the said arrivals to be scheduled and tracked. Through the data_management module, data collection has been streamlined to calculate base metrics such as revenue, average wait, and customer throughput, as well as exporting the said data to a csv file for further analysis.

Key accomplishments since M1 include the completion and integration of the customer arrival process, as well as the establishment of a project structure. The data collection flow has also been implemented, allowing for baseline analytics to be processed. Challenges I encountered revolved around the modularity of the project and working with SimPy. Figuring out how to organize my files in a way that I can expand on easily was challenging, but with some trial and error as well as logically working through components, once I had a structure set up the rest of the development process went smoothly.

My next steps are to create a GUI that allows flexibility and user control over variables, such as the intensity of customer arrival. I also plan to create graphs to show the analytics of customer arrival and revenue. The simulation needs to be iterated over several days for the Schedule Optimizer soon to be created to work with, as it would not be able to optimize much off of one day.